Engi's Farm

Generated by Doxygen 1.8.16

1 Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

2 Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

3 File Index

3.1 File List

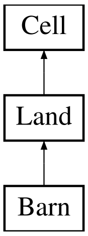
Here is a list of all files with brief descriptions:

4 Class Documentation

4.1 Barn Class Reference

```
#include <Barn.h>
```

Inheritance diagram for Barn:



4.1.1 *

Public Member Functions

- Category getCategory () const

4.1.2 *

Static Private Attributes

- static constexpr Category category {BARN}

4.1.3 *

Additional Inherited Members

4.1.4 Member Function Documentation

@ifstar

getCategory()

4.1.4.1 getCategory()

```
Category Barn::getCategory ( ) const  [virtual]
```

Return kategori dari objek ini

Implements Cell.

4.1.5 Member Data Documentation

@ifstar

category

4.1.5.1 category

```
constexpr Category Barn::category {BARN}  [static], [constexpr], [private]
```

Menandakan bahwa land bertipe Barn

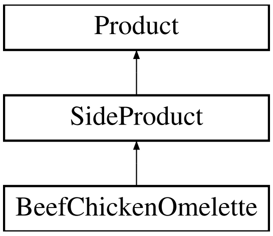
The documentation for this class was generated from the following file:

- Cell/Barn.h

4.2 BeefChickenOmelette Class Reference

```
#include <BeefChickenOmelette.h>
```

Inheritance diagram for BeefChickenOmelette:



4.2.1 *

Public Member Functions

- BeefChickenOmelette ()
- int getPrice () const
- Category getCategory () const

4.2.2 *

Static Public Member Functions

- static LinkedList< Product ∗ > & getRecipe ()

4.2.3 *

Static Private Attributes

- static constexpr int price {250000}
- static constexpr Category category {BEEFCHICKENOMELETTE}
- static LinkedList< Product ∗ > recipe

4.2.4 *

Additional Inherited Members

4.2.5 Constructor & Destructor Documentation

@ifstar

BeefChickenOmelette()

4.2.5.1 BeefChickenOmelette()

```
BeefChickenOmelette::BeefChickenOmelette ( )
```

Constructor untuk inisialisasi recipe

4.2.6 Member Function Documentation

@ifstar

getCategory()

4.2.6.1 getCategory()

```
Category BeefChickenOmelette::getCategory ( ) const  [virtual]
```

Mengembalikan category dari produk

Implements Product.

@ifstar

getPrice()

4.2.6.2 getPrice()

```
int BeefChickenOmelette::getPrice ( ) const  [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements Product.

@ifstar

getRecipe()

4.2.6.3 getRecipe()

```
static LinkedList<Product*>& BeefChickenOmelette::getRecipe ( )  [static]
```

Mengembalikan resep dari produk

4.2.7 Member Data Documentation

@ifstar

category

4.2.7.1 category

```
constexpr Category BeefChickenOmelette::category {BEEFCHICKENOMELETTE}  [static], [constexpr],
[private]
```

Kategori dari BeefChickenOmelette @ifstar

price

4.2.7.2 price

```
constexpr int BeefChickenOmelette::price {250000}  [static], [constexpr], [private]
```

Harga dari BeefChickenOmelette @ifstar

recipe

4.2.7.3 recipe

```
LinkedList<Product*> BeefChickenOmelette::recipe  [static], [private]
```

Resep BeefChickenOmelette. Terdiri dari CowMeat dan ChickenEgg.

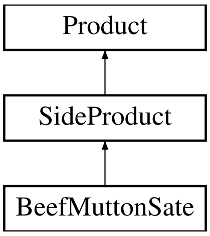
The documentation for this class was generated from the following file:

- Product/BeefChickenOmelette.h

4.3 BeefMuttonSate Class Reference

```
#include <BeefMuttonSate.h>
```

Inheritance diagram for BeefMuttonSate:



4.3.1 *

Public Member Functions

- BeefMuttonSate ()
- int getPrice () const
- Category getCategory () const

4.3.2 *

Static Public Member Functions

- static LinkedList< Product ∗ > & getRecipe ()

4.3.3 *

Static Private Attributes

- static const int price {404000}
- static constexpr Category category {BEEFMUTONSATE}
- static LinkedList< Product ∗ > recipe

4.3.4 *

Additional Inherited Members

4.3.5 Constructor & Destructor Documentation

@ifstar

BeefMuttonSate()

4.3.5.1 BeefMuttonSate()

```
BeefMuttonSate::BeefMuttonSate ( )
```

Constructor untuk inisialisasi recipe

4.3.6 Member Function Documentation

@ifstar

getCategory()

4.3.6.1 getCategory()

```
Category BeefMuttonSate::getCategory ( ) const  [virtual]
```

Mengembalikan category dari produk

Implements Product.

@ifstar

getPrice()

4.3.6.2 getPrice()

```
int BeefMuttonSate::getPrice ( ) const  [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements Product.

@ifstar

getRecipe()

4.3.6.3 getRecipe()

```
static LinkedList<Product*>& BeefMuttonSate::getRecipe ( )  [static]
```

Mengembalikan resep dari produk

4.3.7 Member Data Documentation

@ifstar

category

4.3.7.1 category

```
constexpr Category BeefMuttonSate::category {BEEFMUTONSATE}  [static], [constexpr], [private]
```

Kategori dari BeefMuttonSate @ifstar

price

4.3.7.2 price

```
const int BeefMuttonSate::price {404000}  [static], [private]
```

Harga dari BeefMuttonSate @ifstar

recipe

4.3.7.3 recipe

```
LinkedList<Product*> BeefMuttonSate::recipe  [static], [private]
```

Resep BeefMuttonSate. Terdiri dari CowMeat dan SheepMeat.

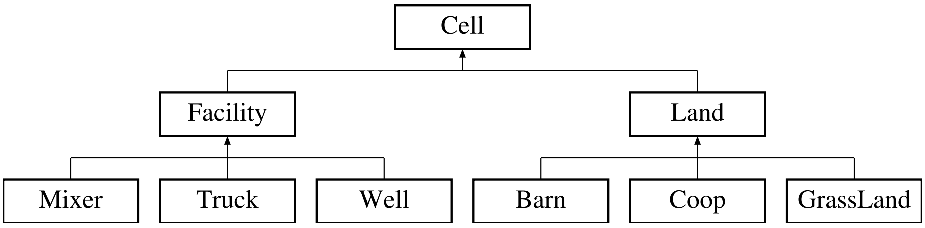
The documentation for this class was generated from the following file:

- Product/BeefMuttonSate.h

4.4 Cell Class Reference

```
#include <Cell.h>
```

Inheritance diagram for Cell:



4.4.1 *

Public Types

- enum Category {
WELL, MIXER, TRUCK, COOP,
GRASSLAND, BARN }

4.4.2 *

Public Member Functions

- virtual ∼Cell ()=0
- virtual bool isFacility () const =0
- virtual Category getCategory () const =0
- bool getIsOcupied ()
- void setIsOcupied (bool)
- virtual void growGrass ()
- virtual void removeGrass ()
- virtual bool isGrassExist () const =0

4.4.3 *

Private Attributes

- bool isOcupied {false}

4.4.4 Member Enumeration Documentation

@ifstar

Category

4.4.4.1 Category

```
enum Cell::Category
```

Jenis kategori Cell

Enumerator

WELL	
MIXER	
TRUCK	
COOP	
GRASSLAND	
BARN	

4.4.5 Constructor & Destructor Documentation

@ifstar

~Cell()

4.4.5.1 ~Cell()

```
virtual Cell::~Cell ( )  [pure virtual]
```

dtor untuk Cell

4.4.6 Member Function Documentation

@ifstar

getCategory()

4.4.6.1 getCategory()

```
virtual Category Cell::getCategory ( ) const  [pure virtual]
```

Return categori dari objek kategori

Implemented in Barn, Coop, GrassLand, Mixer, Truck, and Well.

@ifstar

getIsOcupied()

4.4.6.2 getIsOcupied()

```
bool Cell::getIsOcupied ( )
```

Mengambil nilai boolean isOcupied @ifstar

growGrass()

4.4.6.3 growGrass()

```
virtual void Cell::growGrass ( )  [virtual]
```

Menambah air pada cell. Jika bertipe Land akan menumbuhkan rumput. Jika tidak, tidak akan berefek apa-apa.

Reimplemented in Land.

@ifstar

isFacility()

4.4.6.4 isFacility()

```
virtual bool Cell::isFacility ( ) const  [pure virtual]
```

Return true jika objek adalah Facility

Implemented in Facility, and Land.

@ifstar

isGrassExist()

4.4.6.5 isGrassExist()

```
virtual bool Cell::isGrassExist ( ) const  [pure virtual]
```

Mengembalikan keberadaan grass jika Cell bertipe Land

Implemented in Land, and Facility.

@ifstar

removeGrass()

4.4.6.6 removeGrass()

```
virtual void Cell::removeGrass ( )  [virtual]
```

Reimplemented in Land.

@ifstar

setIsOcupied()

4.4.6.7 setIsOcupied()

```
void Cell::setIsOcupied (
             bool  )
```

Mengganti nilai boolean isOcupied

4.4.7 Member Data Documentation

@ifstar

isOcupied

4.4.7.1 isOcupied

```
bool Cell::isOcupied {false}  [private]
```

Flag yang menandakan cell ditempati oleh sesuatu (Player/FarmAnimal/Facility) atau tidak. True bila cell sedang ditempati oleh sesuatu.

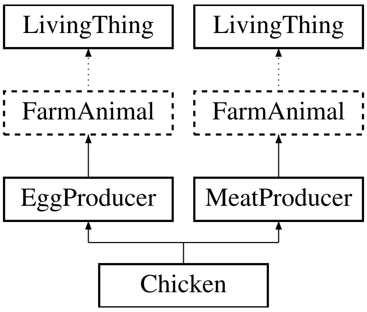
The documentation for this class was generated from the following file:

- Cell/Cell.h

4.5 Chicken Class Reference

```
#include <Chicken.h>
```

Inheritance diagram for Chicken:



4.5.1 *

Public Member Functions

- Chicken (Point position, Cell ∗∗∗&worldMap, int nRowCell, int nCollumnCell)
- FarmProduct ∗ ProduceProduct (Action) const
- std::string makeNoise () const

4.5.2 *

Private Member Functions

- virtual bool canMoveTo (Cell toWhere) const

4.5.3 *

Static Private Attributes

- static constexpr int maxTimeToGetHungryChicken {15}

4.5.4 *

Additional Inherited Members

4.5.5 Constructor & Destructor Documentation

@ifstar

Chicken()

4.5.5.1 Chicken()

```
Chicken::Chicken (
             Point position,
             Cell ***& worldMap,
             int nRowCell,
             int nCollumnCell )
```

Constructor

4.5.6 Member Function Documentation

@ifstar

canMoveTo()

4.5.6.1 canMoveTo()

```
virtual bool Chicken::canMoveTo (
             Cell toWhere ) const  [private], [virtual]
```

Mengecek apakah bisa pindah (tidak out of bound, bertipe Coop atau GrassLand, tidak ada hewan lain)

Reimplemented from EggProducer.

@ifstar

makeNoise()

4.5.6.2 makeNoise()

```
std::string Chicken::makeNoise ( ) const  [virtual]
```

Mengembalikan suara dari Chicken

Implements FarmAnimal.

@ifstar

ProduceProduct()

4.5.6.3 ProduceProduct()

```
FarmProduct* Chicken::ProduceProduct (
             Action  ) const
```

Mengembalikan FarmProduk yang akan dihasilkan Chicken bila Chicken di kill

4.5.7 Member Data Documentation

@ifstar

maxTimeToGetHungryChicken

4.5.7.1 maxTimeToGetHungryChicken

```
constexpr int Chicken::maxTimeToGetHungryChicken {15}  [static], [constexpr], [private]
```

Nilai dari maxTimeToGetHungry

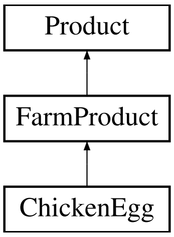
The documentation for this class was generated from the following file:

- FarmAnimal/Chicken.h

4.6 ChickenEgg Class Reference

```
#include <ChickenEgg.h>
```

Inheritance diagram for ChickenEgg:



4.6.1 *

Public Member Functions

- int getPrice () const
- Category getCategory () const

4.6.2 *

Static Private Attributes

- static const int price {2000}
- static constexpr Category category {CHICKENEGG}

4.6.3 *

Additional Inherited Members

4.6.4 Member Function Documentation

@ifstar

getCategory()

4.6.4.1 getCategory()

```
Category ChickenEgg::getCategory ( ) const  [virtual]
```

Mengembalikan category dari produk

Implements Product.

@ifstar

getPrice()

4.6.4.2 getPrice()

```
int ChickenEgg::getPrice ( ) const  [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements Product.

4.6.5 Member Data Documentation

@ifstar

category

4.6.5.1 category

```
constexpr Category ChickenEgg::category {CHICKENEGG}  [static], [constexpr], [private]
```

Kategori ChickenEgg @ifstar

price

4.6.5.2 price

```
const int ChickenEgg::price {2000}  [static], [private]
```

Harga dari ChickenEgg

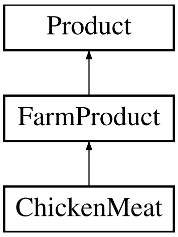
The documentation for this class was generated from the following file:

- Product/ChickenEgg.h

4.7 ChickenMeat Class Reference

```
#include <ChickenMeat.h>
```

Inheritance diagram for ChickenMeat:



4.7.1 *

Public Member Functions

- int getPrice () const
- Category getCategory () const

4.7.2 *

Static Private Attributes

- static const int price {20000}
- static constexpr Category category {CHICKENMEAT}

4.7.3 *

Additional Inherited Members

4.7.4 Member Function Documentation

@ifstar

getCategory()

4.7.4.1 getCategory()

```
Category ChickenMeat::getCategory ( ) const  [virtual]
```

Mengembalikan category dari produk

Implements Product.

@ifstar

getPrice()

4.7.4.2 getPrice()

```
int ChickenMeat::getPrice ( ) const  [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements Product.

4.7.5 Member Data Documentation

@ifstar

category

4.7.5.1 category

```
constexpr Category ChickenMeat::category {CHICKENMEAT}  [static], [constexpr], [private]
```

Kategori dari ChickenMeat @ifstar

price

4.7.5.2 price

```
const int ChickenMeat::price {20000}  [static], [private]
```

Harga dari ChickenMeat

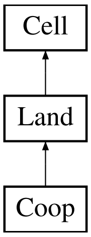
The documentation for this class was generated from the following file:

- Product/ChickenMeat.h

4.8 Coop Class Reference

```
#include <Coop.h>
```

Inheritance diagram for Coop:



4.8.1 *

Public Member Functions

- Category getCategory () const

4.8.2 *

Static Private Attributes

- static constexpr Category category {COOP}

4.8.3 *

Additional Inherited Members

4.8.4 Member Function Documentation

@ifstar

getCategory()

4.8.4.1 getCategory()

```
Category Coop::getCategory ( ) const  [virtual]
```

Return kategori dari objek ini

Implements Cell.

4.8.5 Member Data Documentation

@ifstar

category

4.8.5.1 category

```
constexpr Category Coop::category {COOP}  [static], [constexpr], [private]
```

Menandakan bahwa land bertipe Coop

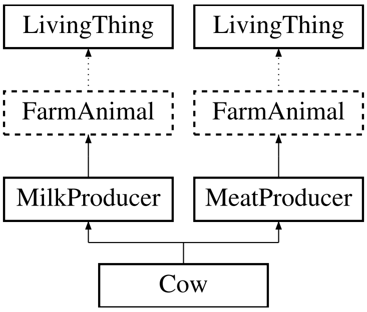
The documentation for this class was generated from the following file:

- Cell/Coop.h

4.9 Cow Class Reference

```
#include <Cow.h>
```

Inheritance diagram for Cow:



4.9.1 *

Public Member Functions

- Cow (Point position, Cell ∗∗∗&worldMap, int nRowCell, int nCollumnCell)
- FarmProduct ∗ ProduceProduct (Action) const
- std::string makeNoise () const

4.9.2 *

Private Member Functions

- virtual bool canMoveTo (Cell toWhere) const

4.9.3 *

Static Private Attributes

- static constexpr int maxTimeToGetHungryCow {20}

4.9.4 *

Additional Inherited Members

4.9.5 Constructor & Destructor Documentation

@ifstar

Cow()

4.9.5.1 Cow()

```
Cow::Cow (
            Point position,
            Cell ***& worldMap,
            int nRowCell,
            int nCollumnCell )
```

Constructor

4.9.6 Member Function Documentation

@ifstar

canMoveTo()

4.9.6.1 canMoveTo()

```
virtual bool Cow::canMoveTo (
            Cell toWhere ) const  [private], [virtual]
```

Mengecek apakah bisa pindah (tidak out of bound, bertipe Barn atau GrassLand, tidak ada hewan lain)

Reimplemented from MeatProducer.

@ifstar

makeNoise()

4.9.6.2 makeNoise()

```
std::string Cow::makeNoise ( ) const  [virtual]
```

Mengembalikan suara dari Cow

Implements FarmAnimal.

@ifstar

ProduceProduct()

4.9.6.3 ProduceProduct()

```
FarmProduct* Cow::ProduceProduct (
            Action  ) const
```

Mengembalikan FarmProduk yang akan dihasilkan Cow bila Cow di kill

4.9.7 Member Data Documentation

@ifstar

maxTimeToGetHungryCow

4.9.7.1 maxTimeToGetHungryCow

```
constexpr int Cow::maxTimeToGetHungryCow {20}  [static], [constexpr], [private]
```

Nilai dari maxTimeToGetHungry

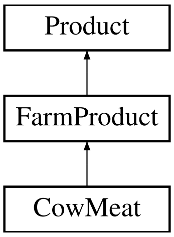
The documentation for this class was generated from the following file:

- FarmAnimal/Cow.h

4.10 CowMeat Class Reference

```
#include <CowMeat.h>
```

Inheritance diagram for CowMeat:



4.10.1 *

Public Member Functions

- int getPrice () const
- Category getCategory () const

4.10.2 *

Static Private Attributes

- static const int price {200000}
- static constexpr Category category {COWMEAT}

4.10.3 *

Additional Inherited Members

4.10.4 Member Function Documentation

@ifstar

getCategory()

4.10.4.1 getCategory()

```
Category CowMeat::getCategory ( ) const  [virtual]
```

Mengembalikan category dari produk

Implements Product.

@ifstar

getPrice()

4.10.4.2 getPrice()

```
int CowMeat::getPrice ( ) const  [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements Product.

4.10.5 Member Data Documentation

@ifstar

category

4.10.5.1 category

```
constexpr Category CowMeat::category {COWMEAT}  [static], [constexpr], [private]
```

Kategori dari CowMeat @ifstar

price

4.10.5.2 price

```
const int CowMeat::price {200000}  [static], [private]
```

Harga dari CowMeat

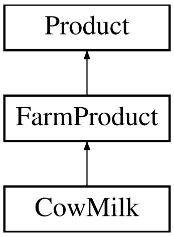
The documentation for this class was generated from the following file:

- Product/CowMeat.h

4.11 CowMilk Class Reference

```
#include <CowMilk.h>
```

Inheritance diagram for CowMilk:



4.11.1 *

Public Member Functions

- int getPrice () const
- Category getCategory () const

4.11.2 *

Static Private Attributes

- static const int price {15000}
- static constexpr Category category {COWMEAT}

4.11.3 *

Additional Inherited Members

4.11.4 Member Function Documentation

@ifstar

getCategory()

4.11.4.1 getCategory()

```
Category CowMilk::getCategory ( ) const  [virtual]
```

Mengembalikan category dari produk

Implements Product.

@ifstar

getPrice()

4.11.4.2 getPrice()

```
int CowMilk::getPrice ( ) const  [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements Product.

4.11.5 Member Data Documentation

@ifstar

category

4.11.5.1 category

```
constexpr Category CowMilk::category {COWMEAT}  [static], [constexpr], [private]
```

Kategori dari CowMilk @ifstar

price

4.11.5.2 price

```
const int CowMilk::price {15000}  [static], [private]
```

Harga dari CowMilk

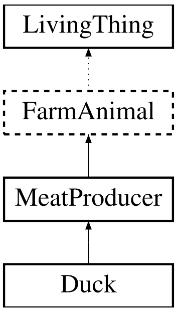
The documentation for this class was generated from the following file:

- Product/CowMilk.h

4.12 Duck Class Reference

```
#include <Duck.h>
```

Inheritance diagram for Duck:



4.12.1 *

Public Member Functions

- Duck (Point position, Cell ∗∗∗&worldMap, int nRowCell, int nCollumnCell)
- FarmProduct ∗ ProduceProduct (Action) const
- std::string makeNoise () const

4.12.2 *

Static Private Attributes

- static constexpr int maxTimeToGetHungryDuck {15}

4.12.3 *

Additional Inherited Members

4.12.4 Constructor & Destructor Documentation

@ifstar

Duck()

4.12.4.1 Duck()

```
Duck::Duck (
            Point position,
            Cell ***& worldMap,
            int nRowCell,
            int nCollumnCell )
```

Constructor

4.12.5 Member Function Documentation

@ifstar

makeNoise()

4.12.5.1 makeNoise()

```
std::string Duck::makeNoise ( ) const  [virtual]
```

Mengembalikan suara dari Duck

Implements FarmAnimal.

@ifstar

ProduceProduct()

4.12.5.2 ProduceProduct()

```
FarmProduct* Duck::ProduceProduct (
             Action  ) const
```

Mengembalikan FarmProduk yang akan dihasilkan Duck bila Duck di kill

4.12.6 Member Data Documentation

@ifstar

maxTimeToGetHungryDuck

4.12.6.1 maxTimeToGetHungryDuck

```
constexpr int Duck::maxTimeToGetHungryDuck {15}  [static], [constexpr], [private]
```

Nilai dari maxTimeToGetHungry

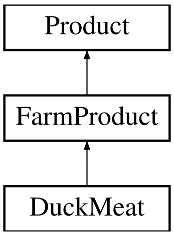
The documentation for this class was generated from the following file:

- FarmAnimal/Duck.h

4.13 DuckMeat Class Reference

```
#include <DuckMeat.h>
```

Inheritance diagram for DuckMeat:



4.13.1 *

Public Member Functions

- int getPrice () const
- Category getCategory () const

4.13.2 *

Static Private Attributes

- static const int price {25000}
- static constexpr Category category {DUCKMEAT}

4.13.3 *

Additional Inherited Members

4.13.4 Member Function Documentation

@ifstar

getCategory()

4.13.4.1 getCategory()

```
Category DuckMeat::getCategory ( ) const  [virtual]
```

Mengembalikan category dari produk

Implements Product.

@ifstar

getPrice()

4.13.4.2 getPrice()

```
int DuckMeat::getPrice ( ) const  [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements Product.

4.13.5 Member Data Documentation

@ifstar

category

4.13.5.1 category

```
constexpr Category DuckMeat::category {DUCKMEAT}  [static], [constexpr], [private]
```

Kategori dari DuckMeat @ifstar

price

4.13.5.2 price

```
const int DuckMeat::price {25000}  [static], [private]
```

Harga dari DuckMeat

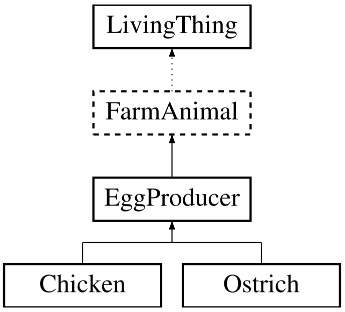
The documentation for this class was generated from the following file:

- Product/DuckMeat.h

4.14 EggProducer Class Reference

```
#include <EggProducer.h>
```

Inheritance diagram for EggProducer:



4.14.1 *

Public Member Functions

- EggProducer (int _maxTimeToGetHungry, Point position, Cell ∗∗∗&worldMap, int nRowCell, int nCollumnCell)
- virtual ∼EggProducer ()=0

4.14.2 *

Private Member Functions

- void eat ()
- virtual bool canMoveTo (Cell toWhere) const

4.14.3 *

Private Attributes

- bool canProduce {false}

4.14.4 *

Additional Inherited Members

4.14.5 Constructor & Destructor Documentation

@ifstar

EggProducer()

4.14.5.1 EggProducer()

```
EggProducer::EggProducer (
            int _maxTimeToGetHungry,
            Point position,
            Cell ***& worldMap,
            int nRowCell,
            int nCollumnCell )
```

Constructor maxTimeToGetHungry dengan nilai H @ifstar

∼EggProducer()

4.14.5.2 ∼EggProducer()

```
virtual EggProducer::∼EggProducer ( )  [pure virtual]
```

Penerusan overloading (virtual) destruktor

4.14.6 Member Function Documentation

@ifstar

canMoveTo()

4.14.6.1 canMoveTo()

```
virtual bool EggProducer::canMoveTo (
            Cell toWhere ) const  [private], [virtual]
```

Mengecek apakah bisa pindah (tidak out of bound, bertipe Coop, tidak ada hewan lain)

Reimplemented in Chicken.

@ifstar

eat()

4.14.6.2 eat()

```
void EggProducer::eat ( )  [private], [virtual]
```

Mengubah nilai canProduce

Reimplemented from FarmAnimal.

4.14.7 Member Data Documentation

@ifstar

canProduce

4.14.7.1 canProduce

```
bool EggProducer::canProduce {false}  [private]
```

Menentukan apakah FarmAnimal dapat menghasilkan produk apabila diinteract

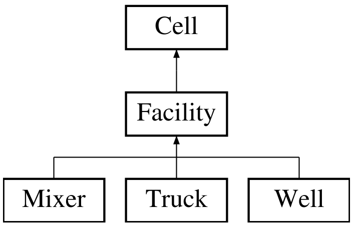
The documentation for this class was generated from the following file:

- FarmAnimal/EggProducer.h

4.15 Facility Class Reference

```
#include <Facility.h>
```

Inheritance diagram for Facility:



4.15.1 *

Public Member Functions

- virtual ∼Facility ()=0
- bool isFacility () const
- bool isGrassExist () const

4.15.2 *

Static Private Attributes

- static constexpr bool facility {true}

4.15.3 *

Additional Inherited Members

4.15.4 Constructor & Destructor Documentation

@ifstar

∼Facility()

4.15.4.1 ∼Facility()

```
virtual Facility::∼Facility ( )  [pure virtual]
```

Destructor Land

4.15.5 Member Function Documentation

@ifstar

isFacility()

4.15.5.1 isFacility()

```
bool Facility::isFacility ( ) const  [virtual]
```

Return true bila Land adalah sebuah facility

Implements Cell.

@ifstar

isGrassExist()

4.15.5.2 isGrassExist()

```
bool Facility::isGrassExist ( ) const  [virtual]
```

Mengembalikan false

Implements Cell.

4.15.6 Member Data Documentation

@ifstar

facility

4.15.6.1 facility

```
constexpr bool Facility::facility {true}  [static], [constexpr], [private]
```

Menandakan bahwa facility

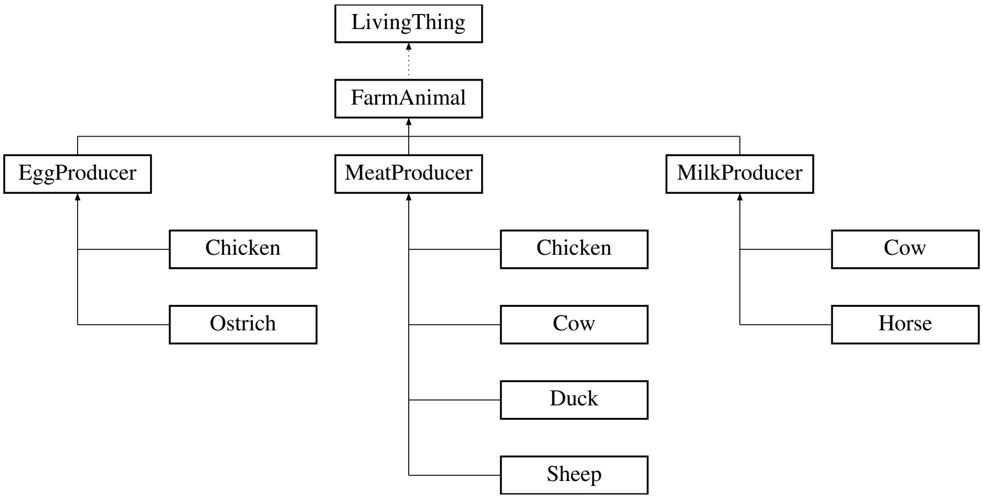
The documentation for this class was generated from the following file:

- Cell/Facility.h

4.16 FarmAnimal Class Reference

```
#include <FarmAnimal.h>
```

Inheritance diagram for FarmAnimal:



4.16.1 *

Public Types

- enum Action { INTERACT, KILL }

4.16.2 *

Public Member Functions

- FarmAnimal (int _maxTimeToGetHungry, Point position, Cell ∗∗∗&worldMap, int nRowCell, int nCollumnCell)
- virtual ∼FarmAnimal ()=0
- void tick ()
- virtual FarmProduct ∗ produceProduct (Action) const =0
- virtual std::string makeNoise () const =0

4.16.3 *

Private Member Functions

- bool isHungry () const
- void decTimeToGetHungry ()
- void decTimetoDeath ()
- bool isDead () const
- virtual void eat ()
- virtual void moveRandomly ()

4.16.4 *

Private Attributes

- int timeToGetHungry
- int timeToDeath
- const int maxTimeToGetHungry

4.16.5 *

Static Private Attributes

- static constexpr int maxTimeToDeath {5}

4.16.6 Member Enumeration Documentation

@ifstar

Action

4.16.6.1 Action

```
enum FarmAnimal::Action
```

Jenis aksi yang dapat dilakukan ke FarmAnimal

Enumerator

INTERACT	
KILL	

4.16.7 Constructor & Destructor Documentation

@ifstar

FarmAnimal()

4.16.7.1 FarmAnimal()

```
FarmAnimal::FarmAnimal (
             int _maxTimeToGetHungry,
             Point position,
             Cell ***& worldMap,
             int nRowCell,
             int nCollumnCell )
```

Constructor maxTimeToGetHungry dengan nilai H @ifstar

∼FarmAnimal()

4.16.7.2 ∼FarmAnimal()

```
virtual FarmAnimal::∼FarmAnimal ( ) [pure virtual]
```

Destructor FarmAnimal

4.16.8 Member Function Documentation

@ifstar

decTimetoDeath()

4.16.8.1 decTimetoDeath()

```
void FarmAnimal::decTimetoDeath ( ) [private]
```

mengurangi timeToDeath @ifstar

decTimeToGetHungry()

4.16.8.2 decTimeToGetHungry()

```
void FarmAnimal::decTimeToGetHungry ( ) [private]
```

mengurangi timeToGetHungry @ifstar

eat()

4.16.8.3 eat()

```
virtual void FarmAnimal::eat ( )  [private], [virtual]
```

Jika FarmAnimal sedang berdiri pada land dengan rumput, maka timeToDeath di set nilai semula dan timeToGdengan nilai sesuai dengan derived classnya, lalu grass di land dihapus

Reimplemented in EggProducer, and MilkProducer.

@ifstar

isDead()

4.16.8.4 isDead()

```
bool FarmAnimal::isDead ( ) const  [private]
```

Mengembalikan true jika timeToDeath == 0, lalu di destruct di main atau di class world @ifstar

isHungry()

4.16.8.5 isHungry()

```
bool FarmAnimal::isHungry ( ) const  [private]
```

return true apabila timeToGetHungry <= 0 @ifstar

makeNoise()

4.16.8.6 makeNoise()

```
virtual std::string FarmAnimal::makeNoise ( ) const  [pure virtual]
```

Mengembalikan suara dari FarmAnimal

Implemented in Chicken, Cow, Duck, Horse, Ostrich, and Sheep.

@ifstar

moveRandomly()

4.16.8.7 moveRandomly()

```
virtual void FarmAnimal::moveRandomly ( )  [private], [virtual]
```

Menggerakan FarmAnimal secara random ke posisi yang mungkin ditempati @ifstar

produceProduct()

4.16.8.8 produceProduct()

```
virtual FarmProduct* FarmAnimal::produceProduct (
             Action  ) const  [pure virtual]
```

Mengembalikan produk yang dihasilkan FarmAnimal apabila diinteract/dikill @ifstar

tick()

4.16.8.9 tick()

```
void FarmAnimal::tick ( )
```

Melakukan aksi yang dilakukan FarmAnimal setiap satuan waktu

4.16.9 Member Data Documentation

@ifstar

maxTimeToDeath

4.16.9.1 maxTimeToDeath

```
constexpr int FarmAnimal::maxTimeToDeath {5} [static], [constexpr], [private]
```

Nilai max dari timeToDeath @ifstar

maxTimeToGetHungry

4.16.9.2 maxTimeToGetHungry

```
const int FarmAnimal::maxTimeToGetHungry [private]
```

Nilai max dari timeToGetHungry @ifstar

timeToDeath

4.16.9.3 timeToDeath

```
int FarmAnimal::timeToDeath [private]
```

Waktu FarmAnimal yang lapar sampai mati Jika tidak lapar, timeToDeath maksimum @ifstar

timeToGetHungry

4.16.9.4 timeToGetHungry

```
int FarmAnimal::timeToGetHungry [private]
```

Waktu FarmAnimal sampai menjadi lapar

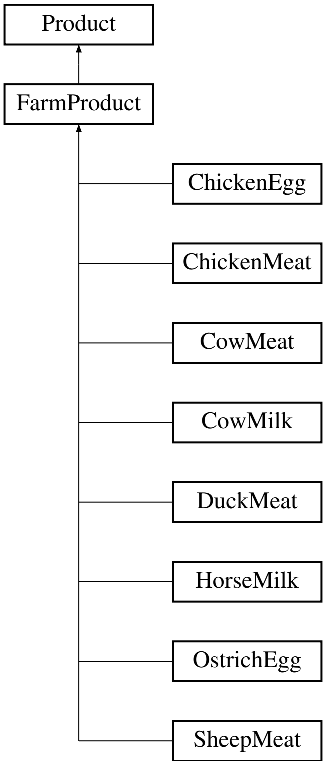
The documentation for this class was generated from the following file:

- FarmAnimal/FarmAnimal.h

4.17 FarmProduct Class Reference

```
#include <FarmProduct.h>
```

Inheritance diagram for FarmProduct:



4.17.1 *

Additional Inherited Members

4.17.2 Detailed Description

Product yang didapat dari hasil interact / kill

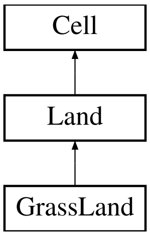
The documentation for this class was generated from the following file:

- Product/FarmProduct.h

4.18 GrassLand Class Reference

```
#include <GrassLand.h>
```

Inheritance diagram for GrassLand:



4.18.1 *

Public Member Functions

- Category getCategory () const

4.18.2 *

Static Private Attributes

- static constexpr Category category {GRASSLAND}

4.18.3 *

Additional Inherited Members

4.18.4 Member Function Documentation

@ifstar

getCategory()

4.18.4.1 getCategory()

```
Category GrassLand::getCategory ( ) const  [virtual]
```

Return kategori dari objek ini

Implements Cell.

4.18.5 Member Data Documentation

@ifstar

category

4.18.5.1 category

```
constexpr Category GrassLand::category {GRASSLAND}  [static], [constexpr], [private]
```

Menandakan bahwa Land ini berkategori GrassLand

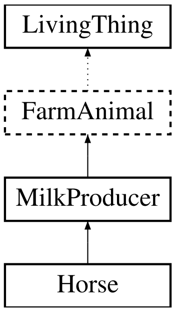
The documentation for this class was generated from the following file:

- Cell/GrassLand.h

4.19 Horse Class Reference

```
#include <Horse.h>
```

Inheritance diagram for Horse:



4.19.1 *

Public Member Functions

- Horse (Point position, Cell ∗∗∗&worldMap, int nRowCell, int nCollumnCell)
- FarmProduct ∗ ProduceProduct (Action) const
- std::string makeNoise () const

4.19.2 *

Static Private Attributes

- static constexpr int maxTimeToGetHungryHorse {18}

4.19.3 *

Additional Inherited Members

4.19.4 Constructor & Destructor Documentation

@ifstar

Horse()

4.19.4.1 Horse()

```
Horse::Horse (
            Point position,
            Cell ***& worldMap,
            int nRowCell,
            int nCollumnCell )
```

Constructor

4.19.5 Member Function Documentation

@ifstar

makeNoise()

4.19.5.1 makeNoise()

```
std::string Horse::makeNoise ( ) const  [virtual]
```

Mengembalikan suara dari Horse

Implements FarmAnimal.

@ifstar

ProduceProduct()

4.19.5.2 ProduceProduct()

```
FarmProduct* Horse::ProduceProduct (
             Action  ) const
```

Mengembalikan FarmProduk yang akan dihasilkan Horse bila Horse di kill

4.19.6 Member Data Documentation

@ifstar

maxTimeToGetHungryHorse

4.19.6.1 maxTimeToGetHungryHorse

```
constexpr int Horse::maxTimeToGetHungryHorse {18}  [static], [constexpr], [private]
```

Nilai dari maxTimeToGetHungry

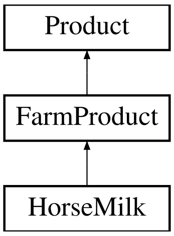
The documentation for this class was generated from the following file:

- FarmAnimal/Horse.h

4.20 HorseMilk Class Reference

```
#include <HorseMilk.h>
```

Inheritance diagram for HorseMilk:



4.20.1 *

Public Member Functions

- int getPrice () const
- Category getCategory () const

4.20.2 *

Static Private Attributes

- static const int price {35000}
- static constexpr Category category {HORSEMILK}

4.20.3 *

Additional Inherited Members

4.20.4 Member Function Documentation

@ifstar

getCategory()

4.20.4.1 getCategory()

```
Category HorseMilk::getCategory ( ) const  [virtual]
```

Mengembalikan category dari produk

Implements Product.

@ifstar

getPrice()

4.20.4.2 getPrice()

```
int HorseMilk::getPrice ( ) const  [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements Product.

4.20.5 Member Data Documentation

@ifstar

category

4.20.5.1 category

```
constexpr Category HorseMilk::category {HORSEMILK}  [static], [constexpr], [private]
```

Kategori dari HorseMilk @ifstar

price

4.20.5.2 price

```
const int HorseMilk::price {35000}  [static], [private]
```

Harga dari HorseMilk

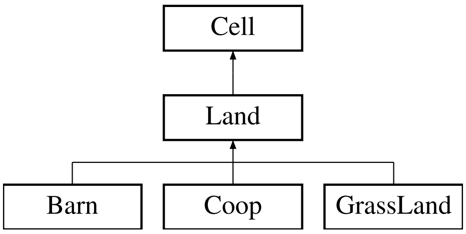
The documentation for this class was generated from the following file:

- Product/HorseMilk.h

4.21 Land Class Reference

```
#include <Land.h>
```

Inheritance diagram for Land:



4.21.1 *

Public Member Functions

- virtual ∼Land ()=0
- bool isFacility () const
- void growGrass ()
- void removeGrass ()
- bool isGrassExist () const

4.21.2 *

Private Attributes

- bool existGrass

4.21.3 *

Static Private Attributes

- static constexpr bool facility {false}

4.21.4 *

Additional Inherited Members

4.21.5 Constructor & Destructor Documentation

@ifstar

∼Land()

4.21.5.1 ∼Land()

```
virtual Land::∼Land ( )  [pure virtual]
```

Destructor Land

4.21.6 Member Function Documentation

@ifstar

growGrass()

4.21.6.1 growGrass()

```
void Land::growGrass ( )  [virtual]
```

Membuat existGrass menjadi true

Reimplemented from Cell.

@ifstar

isFacility()

4.21.6.2 isFacility()

```
bool Land::isFacility ( ) const  [virtual]
```

Return true bila Land adalah sebuah facility

Implements Cell.

@ifstar

isGrassExist()

4.21.6.3 isGrassExist()

```
bool Land::isGrassExist ( ) const  [virtual]
```

Mengembalikan keberadaan grass

Implements Cell.

@ifstar

removeGrass()

4.21.6.4 removeGrass()

```
void Land::removeGrass ( )  [virtual]
```

Reimplemented from Cell.

4.21.7 Member Data Documentation

@ifstar

existGrass

4.21.7.1 existGrass

```
bool Land::existGrass  [private]
```

Flag yang menandakan apakah terdapat rumput diatas suatu cell atau tidak @ifstar

facility

4.21.7.2 facility

```
constexpr bool Land::facility {false}  [static], [constexpr], [private]
```

Menandakan bahwa land bukan facility

The documentation for this class was generated from the following file:

- Cell/Land.h

4.22 LinkedList< T > Class Template Reference

```
#include <LinkedList.h>
```

4.22.1 *

Public Member Functions

- LinkedList ()
- LinkedList (std::initializer_list< T > args)
- LinkedList (const LinkedList< T > &l)
- ∼LinkedList ()
- LinkedList< T > & operator= (const LinkedList< T > &l)
- int find (T elm)
- bool isEmpty () const
- void add (T elm)
- void remove (T elm)
- void removeIdx (int idx)
- T get (int idx)
- T & operator[ ] (int idx)

4.22.2 *

Private Attributes

- LinkedListNode< T > ∗ list

4.22.3 Detailed Description

template<class T>
class LinkedList< T >

Tipe data LinkedList, diimplementasi secara rekursif dengan LinkedListNode

4.22.4 Constructor & Destructor Documentation

@ifstar

LinkedList() [1/3]

4.22.4.1 LinkedList() [1/3]

```
template<class T >
LinkedList< T >::LinkedList ( )
```

Konstruktor default LinkedList, membuat empty list @ifstar

LinkedList() [2/3]

4.22.4.2 LinkedList() [2/3]

```
template<class T>
LinkedList< T >::LinkedList (
             std::initializer_list< T > args )
```

Konstruktor dengan initializer list @ifstar

LinkedList() [3/3]

4.22.4.3 LinkedList() [3/3]

```
template<class T>
LinkedList< T >::LinkedList (
             const LinkedList< T > & l )
```

Copy constructor LinkedList @ifstar

∼LinkedList()

4.22.4.4 ∼LinkedList()

```
template<class T >
LinkedList< T >::∼LinkedList ( )
```

Destructor LinkedList

4.22.5 Member Function Documentation

@ifstar

add()

4.22.5.1 add()

```
template<class T>
void LinkedList< T >::add (
             T elm )
```

Menambah elm sebagai elemen terakhir @ifstar

find()

4.22.5.2 find()

```
template<class T>
int LinkedList< T >::find (
             T elm )
```

Mencari indeks pertama dari elm dari LinkedList. Jika tidak ada, bernilai -1. @ifstar

get()

4.22.5.3 get()

```
template<class T >
T LinkedList< T >::get (
             int idx )
```

Mengembalikan elemen berindeks idx. Jika diluar range, melempar "Index is out of bounds". @ifstar

isEmpty()

4.22.5.4 isEmpty()

```
template<class T >
bool LinkedList< T >::isEmpty ( ) const
```

Mengembalikan apakah list empty atau tidak @ifstar

operator=()

4.22.5.5 operator=()

```
template<class T>
LinkedList< T > & LinkedList< T >::operator= (
             const LinkedList< T > & l )
```

Operator= LinkedList @ifstar

operator[]()

4.22.5.6 operator[]()

```
template<class T >
T & LinkedList< T >::operator[] (
             int idx )
```

Mengembalikan reference ke elemen berindeks idx. Jika diluar range, melempar "Index is out of bounds". @ifstar

remove()

4.22.5.7 remove()

```
template<class T>
void LinkedList< T >::remove (
             T elm )
```

Menghapus keberadaan pertama elm Membuat list temp berisi tail untuk dipindahkan ke list sekarang @ifstar

removeIdx()

4.22.5.8 removeIdx()

```
template<class T >
void LinkedList< T >::removeIdx (
             int idx )
```

Menghapus elemen berindeks idx. Jika diluar range, melempar "Index is out of bounds". Membuat list temp berisi tail untuk dipindahkan ke list sekarang

4.22.6 Member Data Documentation

@ifstar

list

4.22.6.1 list

```
template<class T>
LinkedListNode<T>* LinkedList< T >::list  [private]
```

Pointer ke LinkedListNode, kalau empty bernilai nullptr

The documentation for this class was generated from the following file:

- LinkedList.h

4.23 LinkedListNode< T > Class Template Reference

```
#include <LinkedList.h>
```

4.23.1 *

Public Member Functions

- LinkedListNode (T head, LinkedList< T > tail)

4.23.2 *

Public Attributes

- friend LinkedList< T >

4.23.3 *

Private Attributes

- T head
- LinkedList< T > tail

4.23.4 Detailed Description

template<class T>
class LinkedListNode< T >

Forward declaration dari kelas LinkedListNode

Anggota kelas implementasi LinkedList secara rekursifs

4.23.5 Constructor & Destructor Documentation

@ifstar

LinkedListNode()

4.23.5.1 LinkedListNode()

```
template<class T>
LinkedListNode< T >::LinkedListNode (
             T head,
             LinkedList< T > tail )
```

Konstruktor LinkedListNode dengan initializer list

4.23.6 Member Data Documentation

@ifstar

head

4.23.6.1 head

```
template<class T>
T LinkedListNode< T >::head  [private]
```

Tipe data pertama pada LinkedListNode   @ifstar

LinkedList< T >

4.23.6.2 LinkedList< T >

```
template<class T>
friend LinkedListNode< T >::LinkedList< T >
```

Membuat LinkedList dapat mengakses head dan tail   @ifstar

tail

4.23.6.3 tail

```
template<class T>
LinkedList<T> LinkedListNode< T >::tail  [private]
```

Sisa dari LinkedListNode berupa LinkedList

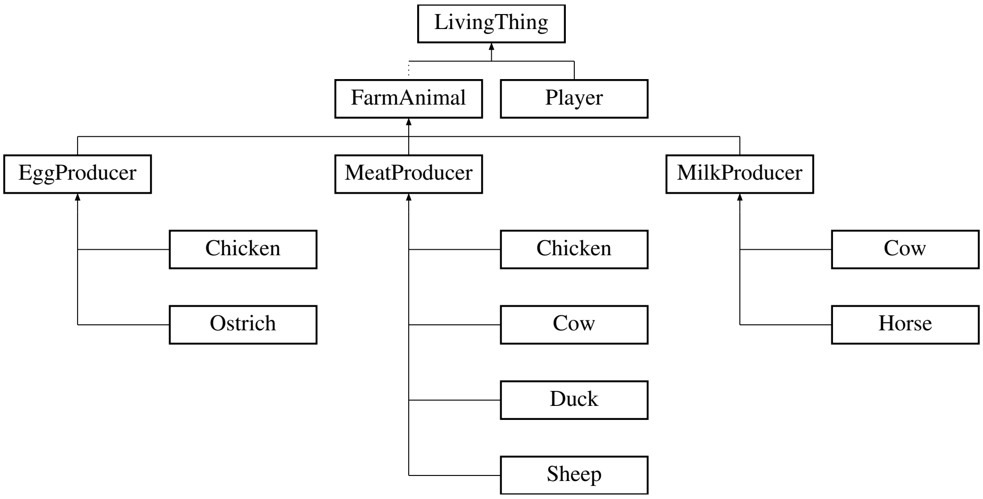
The documentation for this class was generated from the following file:

- LinkedList.h

4.24 LivingThing Class Reference

```
#include <LivingThing.h>
```

Inheritance diagram for LivingThing:



4.24.1 *

Public Member Functions

- LivingThing (Point position, Cell ∗∗∗&worldMap, int nRowCell, int nCollumnCell)
- virtual ∼LivingThing ()=0
- Point getPosition () const
- bool move (Direction toWhere)
- virtual char render ()=0

4.24.2 *

Protected Attributes

- Cell ∗∗∗& worldMap
- int nRowCell
- int nCollumnCell

4.24.3 *

Private Member Functions

- virtual bool canMoveTo (Cell toWhere)=0

4.24.4 *

Private Attributes

- Point position

4.24.5 Constructor & Destructor Documentation

@ifstar

LivingThing()

4.24.5.1 LivingThing()

```
LivingThing::LivingThing (
             Point position,
             Cell ***& worldMap,
             int nRowCell,
             int nCollumnCell )
```

Constructor LivingThing @ifstar

∼LivingThing()

4.24.5.2 ∼LivingThing()

```
virtual LivingThing::∼LivingThing ( )  [pure virtual]
```

Destructor dari LivingThing

4.24.6 Member Function Documentation

@ifstar

canMoveTo()

4.24.6.1 canMoveTo()

```
virtual bool LivingThing::canMoveTo (
             Cell toWhere )  [private], [pure virtual]
```

Apakah bisa masuk suatu area (cek out of bound, jenis Cell, kekosongan Cell)

Implemented in Player.

@ifstar

getPosition()

4.24.6.2 getPosition()

```
Point LivingThing::getPosition ( ) const
```

Mengembalikan position @ifstar

move()

4.24.6.3 move()

```
bool LivingThing::move (
            Direction toWhere )
```

Berpindah ke suatu lokasi. Apabila tidak bisa (!canMoveTo), throw "Cannot move to the direction". @ifstar

render()

4.24.6.4 render()

```
virtual char LivingThing::render ( )  [pure virtual]
```

Mengembalikan char untuk dirender ke layar

Implemented in Player.

4.24.7 Member Data Documentation

@ifstar

nCollumnCell

4.24.7.1 nCollumnCell

```
int LivingThing::nCollumnCell  [protected]
```

Nilai efektif kolom untuk Matriks Cell @ifstar

nRowCell

4.24.7.2 nRowCell

```
int LivingThing::nRowCell  [protected]
```

Nilai efektif baris untuk Matriks Cell @ifstar

position

4.24.7.3 position

```
Point LivingThing::position  [private]
```

Posisi dari LivingThing @ifstar

worldMap

4.24.7.4 worldMap

```
Cell***& LivingThing::worldMap  [protected]
```

Representasi dunia tempat LivingThing tinggal

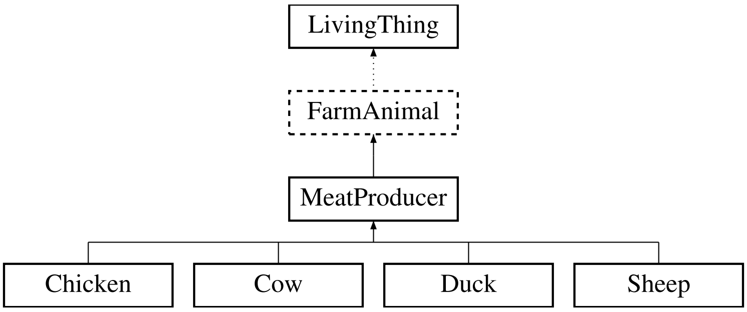
The documentation for this class was generated from the following file:

- LivingThing.h

4.25 MeatProducer Class Reference

```
#include <MeatProducer.h>
```

Inheritance diagram for MeatProducer:



4.25.1 *

Public Member Functions

- MeatProducer (int _maxTimeToGetHungry, Point position, Cell ∗∗∗&worldMap, int nRowCell, int nCollumnCell)
- virtual ∼MeatProducer ()=0

4.25.2 *

Private Member Functions

- virtual bool canMoveTo (Cell toWhere) const

4.25.3 *

Additional Inherited Members

4.25.4 Constructor & Destructor Documentation

@ifstar

MeatProducer()

4.25.4.1 MeatProducer()

```
MeatProducer::MeatProducer (
                int _maxTimeToGetHungry,
                Point position,
                Cell ***& worldMap,
                int nRowCell,
                int nCollumnCell )
```

Constructor maxTimeToGetHungry dengan nilai H @ifstar

∼MeatProducer()

4.25.4.2 ∼MeatProducer()

```
virtual MeatProducer::∼MeatProducer ( )  [pure virtual]
```

Penerusan overloading (virtual) destruktor

4.25.5 Member Function Documentation

@ifstar

canMoveTo()

4.25.5.1 canMoveTo()

```
virtual bool MeatProducer::canMoveTo (
             Cell toWhere ) const  [private], [virtual]
```

Mengecek apakah bisa pindah (tidak out of bound, bertipe GrassLand, tidak ada hewan lain)

Reimplemented in Chicken, and Cow.

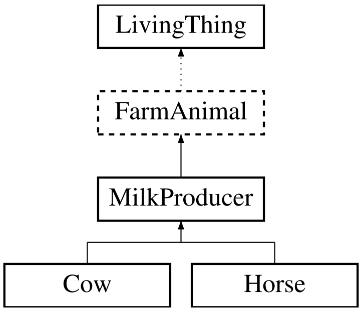
The documentation for this class was generated from the following file:

- FarmAnimal/MeatProducer.h

4.26 MilkProducer Class Reference

```
#include <MilkProducer.h>
```

Inheritance diagram for MilkProducer:



4.26.1 *

Public Member Functions

- MilkProducer (int _maxTimeToGetHungry, Point position, Cell ∗∗∗&worldMap, int nRowCell, int nCollumnCell)
- virtual ∼MilkProducer ()=0

4.26.2 *

Private Member Functions

- void eat ()
- virtual bool canMoveTo (Cell toWhere) const

4.26.3 *

Private Attributes

- bool canProduce {false}

4.26.4 *

Additional Inherited Members

4.26.5 Constructor & Destructor Documentation

@ifstar

MilkProducer()

4.26.5.1 MilkProducer()

```
MilkProducer::MilkProducer (
             int _maxTimeToGetHungry,
             Point position,
             Cell ***& worldMap,
             int nRowCell,
             int nCollumnCell )
```

Constructor maxTimeToGetHungry dengan nilai H @ifstar

∼MilkProducer()

4.26.5.2 ∼MilkProducer()

```
virtual MilkProducer::∼MilkProducer ( )  [pure virtual]
```

Penerusan overloading (virtual) destruktor

4.26.6 Member Function Documentation

@ifstar

canMoveTo()

4.26.6.1 canMoveTo()

```
virtual bool MilkProducer::canMoveTo (
            Cell toWhere ) const  [private], [virtual]
```

Mengecek apakah bisa pindah (tidak out of bound, bertipe Barn, tidak ada hewan lain)

Reimplemented in Cow.

@ifstar

eat()

4.26.6.2 eat()

```
void MilkProducer::eat ( )  [private], [virtual]
```

Mengubah nilai canProduce

Reimplemented from FarmAnimal.

4.26.7 Member Data Documentation

@ifstar

canProduce

4.26.7.1 canProduce

```
bool MilkProducer::canProduce {false}  [private]
```

Menentukan apakah FarmAnimal dapat menghasilkan produk apabila diinteract

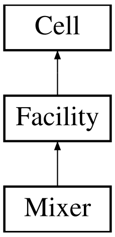
The documentation for this class was generated from the following file:

- FarmAnimal/MilkProducer.h

4.27 Mixer Class Reference

```
#include <Mixer.h>
```

Inheritance diagram for Mixer:



4.27.1 *

Public Member Functions

- Category getCategory () const

4.27.2 *

Static Private Attributes

- static constexpr Category category {MIXER}

4.27.3 *

Additional Inherited Members

4.27.4 Member Function Documentation

@ifstar

getCategory()

4.27.4.1 getCategory()

```
Category Mixer::getCategory ( ) const  [virtual]
```

Return kategori dari objek ini

Implements Cell.

4.27.5 Member Data Documentation

@ifstar

category

4.27.5.1 category

```
constexpr Category Mixer::category {MIXER}  [static], [constexpr], [private]
```

Menandakan bahwa land bertipe Mixer

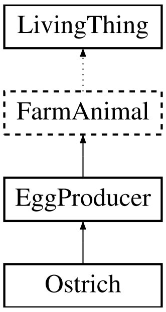
The documentation for this class was generated from the following file:

- Cell/Mixer.h

4.28 Ostrich Class Reference

```
#include <Ostrich.h>
```

Inheritance diagram for Ostrich:



4.28.1 *

Public Member Functions

- Ostrich (Point position, Cell ∗∗∗&worldMap, int nRowCell, int nCollumnCell)
- FarmProduct ∗ ProduceProduct (Action) const
- std::string makeNoise () const

4.28.2 *

Static Private Attributes

- static constexpr int maxTimeToGetHungryOstrich {15}

4.28.3 *

Additional Inherited Members

4.28.4 Constructor & Destructor Documentation

@ifstar

Ostrich()

4.28.4.1 Ostrich()

```
Ostrich::Ostrich (
             Point position,
             Cell ***& worldMap,
             int nRowCell,
             int nCollumnCell )
```

Constructor

4.28.5 Member Function Documentation

@ifstar

makeNoise()

4.28.5.1 makeNoise()

```
std::string Ostrich::makeNoise ( ) const  [virtual]
```

Mengembalikan suara dari Chicken

Implements FarmAnimal.

@ifstar

ProduceProduct()

4.28.5.2 ProduceProduct()

```
FarmProduct* Ostrich::ProduceProduct (
             Action  ) const
```

Mengembalikan FarmProduk yang akan dihasilkan Ostrich bila Ostrich di kill

4.28.6 Member Data Documentation

@ifstar

maxTimeToGetHungryOstrich

4.28.6.1 maxTimeToGetHungryOstrich

```
constexpr int Ostrich::maxTimeToGetHungryOstrich {15}  [static], [constexpr], [private]
```

Nilai dari maxTimeToGetHungry

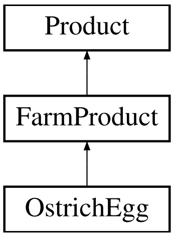
The documentation for this class was generated from the following file:

- FarmAnimal/Ostrich.h

4.29 OstrichEgg Class Reference

```
#include <OstrichEgg.h>
```

Inheritance diagram for OstrichEgg:



4.29.1 *

Public Member Functions

- int getPrice () const
- Category getCategory () const

4.29.2 *

Static Private Attributes

- static const int price {40000}
- static constexpr Category category {OSTRICHEGG}

4.29.3 *

Additional Inherited Members

4.29.4 Member Function Documentation

@ifstar

getCategory()

4.29.4.1 getCategory()

```
Category OstrichEgg::getCategory ( ) const  [virtual]
```

Mengembalikan category dari produk

Implements Product.

@ifstar

getPrice()

4.29.4.2 getPrice()

```
int OstrichEgg::getPrice ( ) const  [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements Product.

4.29.5 Member Data Documentation

@ifstar

category

4.29.5.1 category

```
constexpr Category OstrichEgg::category {OSTRICHEGG}  [static], [constexpr], [private]
```

Kategori dari OstrichEgg @ifstar

price

4.29.5.2 price

```
const int OstrichEgg::price {40000}  [static], [private]
```

Harga dari OstrichEgg

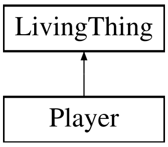
The documentation for this class was generated from the following file:

- Product/OstrichEgg.h

4.30 Player Class Reference

```
#include <Player.h>
```

Inheritance diagram for Player:



4.30.1 *

Public Member Functions

- Player (Point position, Cell ∗∗∗&worldMap, int nRowCell, int nCollumnCell)
- ∼Player ()
- void talk (LinkedList< FarmAnimal > &farmAnimal)
- void interact (LinkedList< FarmAnimal > &farmAnimal)
- void kill (LinkedList< FarmAnimal > &farmAnimal)
- void grow ()
- void mix (Product ∗makeTo)
- char render ()

4.30.2 *

Private Member Functions

- bool canMoveTo (Cell toWhere)

4.30.3 *

Private Attributes

- LinkedList< Product & > inventory
- int money {500000}
- int water {5}

4.30.4 *

Static Private Attributes

- static LinkedList< SideProduct * > recipeBook

4.30.5 *

Additional Inherited Members

4.30.6 Constructor & Destructor Documentation

@ifstar

Player()

4.30.6.1 Player()

```
Player::Player (
            Point position,
            Cell ***& worldMap,
            int nRowCell,
            int nCollumnCell )
```

Constructor Player di position, recipeBook diinisalisasi dengan semua SideProduct yang terdefinisi @ifstar

∼Player()

4.30.6.2 ∼Player()

```
Player::∼Player ( )
```

Destructor Player

4.30.7 Member Function Documentation

@ifstar

canMoveTo()

4.30.7.1 canMoveTo()

```
bool Player::canMoveTo (
            Cell toWhere )  [private], [virtual]
```

Apakah bisa masuk suatu area (cek out of bound, jenis Cell, kekosongan Cell)

Implements LivingThing.

@ifstar

grow()

4.30.7.2 grow()

```
void Player::grow ( )
```

Menumbuhkan rumput pada cell yang sedang ditempati oleh Player @ifstar

interact()

4.30.7.3 interact()

```
void Player::interact (
            LinkedList< FarmAnimal > & farmAnimal )
```

Player mengambil FarmProduct dari semua FarmAnimal terdekat tanpa membunuh FarmAnimal tersebut. Bekerja untuk FarmAnimal jenis MilkProducing dan EggProducing. Contoh FarmProduct : ChickenEgg, CowMilk. @ifstar

kill()

4.30.7.4 kill()

```
void Player::kill (
            LinkedList< FarmAnimal > & farmAnimal )
```

Player mengambil FarmProduct dari semua FarmAnimal terdekat dengan cara membunuh FarmAnimal tersebut. Bekerja untuk FarmAnimal jenis MeatProducing. Contoh FarmProduct : CowMeat, ChickenMeat. @ifstar

mix()

4.30.7.5 mix()

```
void Player::mix (
            Product * makeTo )
```

Menciptakan SideProduct dari FarmProduct bila Player dekat dengan mixer @ifstar render()

4.30.7.6 render()

```
char Player::render ( )  [virtual]
```

Mengembalikan char untuk dirender ke layar

Implements LivingThing.

@ifstar

talk()

4.30.7.7 talk()

```
void Player::talk (
             LinkedList< FarmAnimal > & farmAnimal )
```

Player berbicara dengan semua FarmAnimal terdekat.

4.30.8 Member Data Documentation

@ifstar

inventory

4.30.8.1 inventory

```
LinkedList<Product&> Player::inventory  [private]
```

Product yang dipegang Player @ifstar

money

4.30.8.2 money

```
int Player::money {500000}  [private]
```

Uang yang dimiliki Player @ifstar

recipeBook

4.30.8.3 recipeBook

```
LinkedList<SideProduct*> Player::recipeBook  [static], [private]
```

Digunakan untuk melakukan pengecekan saat melakukan method mix Contoh Pengunaan : Bila player ingin membuat BeefMuttonSate, program tranversal di recipeeBook sampai menemukan sideProdect dengan Category = BEEFMUTTONSATE lalu melihat resep dari objek tersebut. recipeBook diinisalisasi di implementasi @ifstar

water

4.30.8.4 water

```
int Player::water {5}  [private]
```

Air yang dipegang Player

The documentation for this class was generated from the following file:

- Player.h

4.31 Point Struct Reference

```
#include <Point.h>
```

4.31.1 *

Public Attributes

- int x
- int y

4.31.2 Member Data Documentation

@ifstar

x

4.31.2.1 x

```
int Point::x
```

@ifstar

y

4.31.2.2 y

```
int Point::y
```

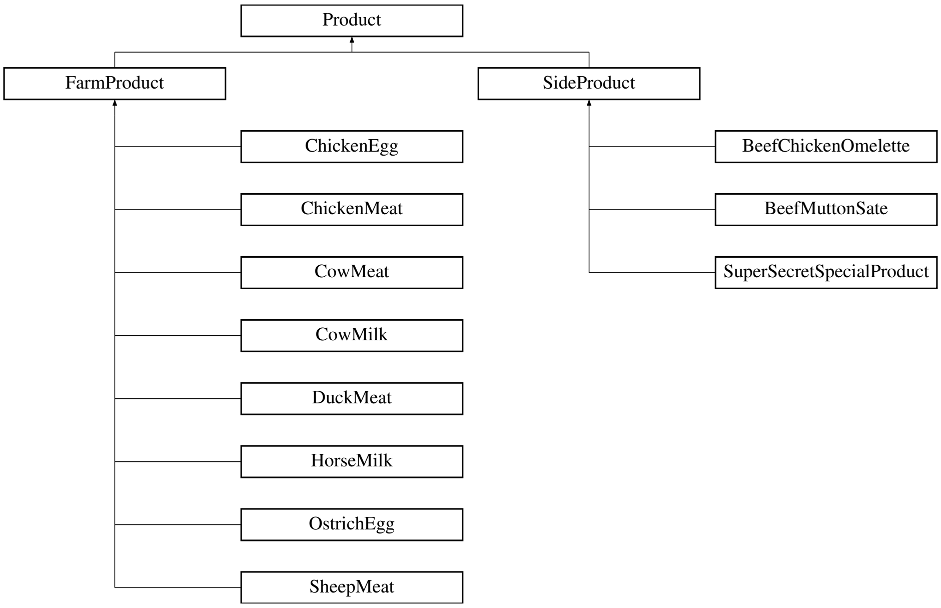
The documentation for this struct was generated from the following file:

- Point.h

4.32 Product Class Reference

```
#include <Product.h>
```

Inheritance diagram for Product:



4.32.1 *

Public Types

- enum Category {
  CHICKENEGG, CHICKENMEAT, BEEFCHICKENOMELETTE, BEEFMUTONSATE,
  COWMILK, COWMEAT, DUCKMEAT, HORSEMILK,
  OSTRICHEGG, SHEEPMEAT, SUPERSECRETSPECIALPRODUCT }

4.32.2 *

Public Member Functions

- virtual int getPrice () const =0
- virtual Category getCategory () const =0

4.32.3 Member Enumeration Documentation

@ifstar

Category

4.32.3.1 Category

```
enum Product::Category
```

Enumerator

CHICKENEGG	
CHICKENMEAT	
BEEFCHICKENOMELETTE	
BEEFMUTONSATE	
COWMILK	
COWMEAT	
DUCKMEAT	
HORSEMILK	
OSTRICHEGG	
SHEEPMEAT	
SUPERSECRETSPECIALPRODUCT	

4.32.4 Member Function Documentation

@ifstar

getCategory()

4.32.4.1 getCategory()

```
virtual Category Product::getCategory ( ) const  [pure virtual]
```

mengembalikan kategori dari produk ini

Implemented in BeefChickenOmelette, BeefMuttonSate, SuperSecretSpecialProduct, ChickenEgg, ChickenMeat, CowMeat, CowMilk, DuckMeat, HorseMilk, OstrichEgg, and SheepMeat.

@ifstar

getPrice()

4.32.4.2 getPrice()

```
virtual int Product::getPrice ( ) const  [pure virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implemented in BeefChickenOmelette, BeefMuttonSate, SuperSecretSpecialProduct, ChickenEgg, ChickenMeat, CowMeat, CowMilk, DuckMeat, HorseMilk, OstrichEgg, and SheepMeat.

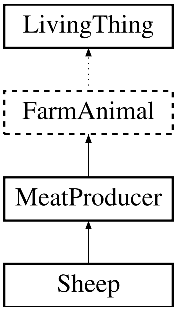
The documentation for this class was generated from the following file:

- Product/Product.h

4.33 Sheep Class Reference

```
#include <Sheep.h>
```

Inheritance diagram for Sheep:



4.33.1 *

Public Member Functions

- Sheep (Point position, Cell ∗∗∗&worldMap, int nRowCell, int nCollumnCell)
- FarmProduct ∗ ProduceProduct (Action) const
- std::string makeNoise () const

4.33.2 *

Static Private Attributes

- static constexpr int maxTimeToGetHungrySheep {15}

4.33.3 *

Additional Inherited Members

4.33.4 Constructor & Destructor Documentation

@ifstar

Sheep()

4.33.4.1 Sheep()

```
Sheep::Sheep (
            Point position,
            Cell ***& worldMap,
            int nRowCell,
            int nCollumnCell )
```

Constructor

4.33.5 Member Function Documentation

@ifstar

makeNoise()

4.33.5.1 makeNoise()

```
std::string Sheep::makeNoise ( ) const  [virtual]
```

Mengembalikan suara dari Sheep

Implements FarmAnimal.

@ifstar

ProduceProduct()

4.33.5.2 ProduceProduct()

```
FarmProduct* Sheep::ProduceProduct (
            Action  ) const
```

Mengembalikan FarmProduk yang akan dihasilkan Sheep bila Sheep di kill

4.33.6 Member Data Documentation

@ifstar

maxTimeToGetHungrySheep

4.33.6.1 maxTimeToGetHungrySheep

```
constexpr int Sheep::maxTimeToGetHungrySheep {15}  [static], [constexpr], [private]
```

Nilai dari maxTimeToGetHungry

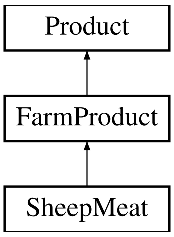
The documentation for this class was generated from the following file:

- FarmAnimal/Sheep.h

4.34 SheepMeat Class Reference

```
#include <SheepMeat.h>
```

Inheritance diagram for SheepMeat:



4.34.1 *

Public Member Functions

- int getPrice () const
- Category getCategory () const

4.34.2 *

Static Private Attributes

- static const int price {100000}
- static constexpr Category category {SHEEPMEAT}

4.34.3 *

Additional Inherited Members

4.34.4 Member Function Documentation

@ifstar

getCategory()

4.34.4.1 getCategory()

```
Category SheepMeat::getCategory ( ) const  [virtual]
```

Mengembalikan category dari produk

Implements Product.

@ifstar

getPrice()

4.34.4.2 getPrice()

```
int SheepMeat::getPrice ( ) const  [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements Product.

4.34.5 Member Data Documentation

@ifstar

category

4.34.5.1 category

```
constexpr Category SheepMeat::category {SHEEPMEAT}  [static], [constexpr], [private]
```

Kategori dari SheepMeat @ifstar

price

4.34.5.2 price

```
const int SheepMeat::price {100000}  [static], [private]
```

Harga dari SheepMeat

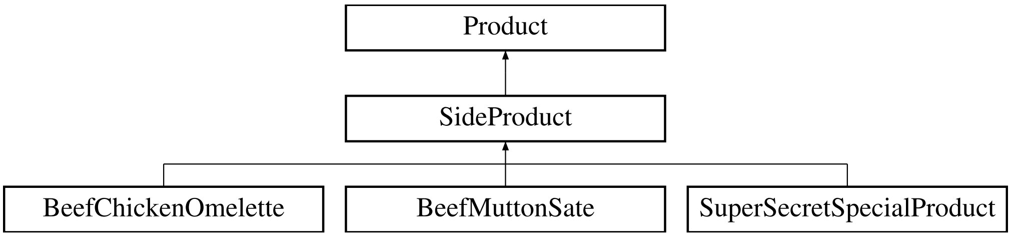
The documentation for this class was generated from the following file:

- Product/SheepMeat.h

4.35 SideProduct Class Reference

```
#include <SideProduct.h>
```

Inheritance diagram for SideProduct:



4.35.1 *

Additional Inherited Members

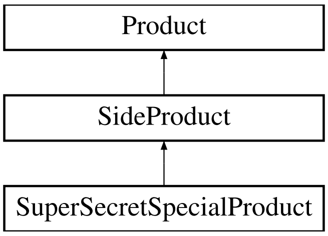
The documentation for this class was generated from the following file:

- Product/SideProduct.h

4.36 SuperSecretSpecialProduct Class Reference

```
#include <SuperSecretSpecialProduct.h>
```

Inheritance diagram for SuperSecretSpecialProduct:



4.36.1 *

Public Member Functions

- SuperSecretSpecialProduct ()
- int getPrice () const
- Category getCategory () const

4.36.2 *

Static Public Member Functions

- static LinkedList< Product * > & getRecipe ()

4.36.3 *

Static Private Attributes

- static const int price {999999}
- static constexpr Category category {SUPERSECRETSPECIALPRODUCT}
- static const LinkedList< Product * > recipe

4.36.4 *

Additional Inherited Members

4.36.5 Constructor & Destructor Documentation

@ifstar

SuperSecretSpecialProduct()

4.36.5.1 SuperSecretSpecialProduct()

```
SuperSecretSpecialProduct::SuperSecretSpecialProduct ( )
```

Constructor untuk inisialisasi recipe

4.36.6 Member Function Documentation

@ifstar

getCategory()

4.36.6.1 getCategory()

```
Category SuperSecretSpecialProduct::getCategory ( ) const  [virtual]
```

Mengembalikan category dari produk

Implements Product.

@ifstar

getPrice()

4.36.6.2 getPrice()

```
int SuperSecretSpecialProduct::getPrice ( ) const  [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements Product.

@ifstar

getRecipe()

4.36.6.3 getRecipe()

```
static LinkedList<Product*>& SuperSecretSpecialProduct::getRecipe ( )  [static]
```

Mengembalikan resep dari produk

4.36.7 Member Data Documentation

@ifstar

category

4.36.7.1 category

```
constexpr Category SuperSecretSpecialProduct::category {SUPERSECRETSPECIALPRODUCT}  [static],
[constexpr], [private]
```

Kategori dari SuperSecretSpecialProduct @ifstar

price

4.36.7.2 price

```
const int SuperSecretSpecialProduct::price {999999}  [static], [private]
```

Harga dari SuperSecretSpecialProduct @ifstar

recipe

4.36.7.3 recipe

```
const LinkedList<Product*> SuperSecretSpecialProduct::recipe  [static], [private]
```

Resep SuperSecretSpecialProduct. Terdiri dari OstrichEgg dan HorseMeat.

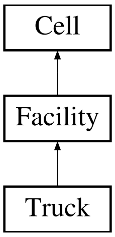
The documentation for this class was generated from the following file:

- Product/SuperSecretSpecialProduct.h

4.37 Truck Class Reference

```
#include <Truck.h>
```

Inheritance diagram for Truck:



4.37.1 *

Public Member Functions

- Category getCategory () const

4.37.2 *

Static Private Attributes

- static constexpr Category category {TRUCK}

4.37.3 *

Additional Inherited Members

4.37.4 Member Function Documentation

@ifstar

getCategory()

4.37.4.1 getCategory()

```
Category Truck::getCategory ( ) const  [virtual]
```

Return kategori dari objek ini

Implements Cell.

4.37.5 Member Data Documentation

@ifstar

category

4.37.5.1 category

```
constexpr Category Truck::category {TRUCK}  [static], [constexpr], [private]
```

Menandakan bahwa land bertipe Truck

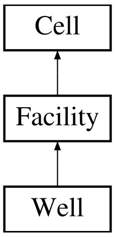
The documentation for this class was generated from the following file:

- Cell/Truck.h

4.38 Well Class Reference

```
#include <Well.h>
```

Inheritance diagram for Well:



4.38.1 *

Public Member Functions

- Category getCategory () const

4.38.2 *

Static Private Attributes

- static constexpr Category category {WELL}

4.38.3 *

Additional Inherited Members

4.38.4 Member Function Documentation

@ifstar

getCategory()

4.38.4.1 getCategory()

```
Category Well::getCategory ( ) const  [virtual]
```

Return kategori dari objek ini

Implements Cell.

4.38.5 Member Data Documentation

@ifstar

category

4.38.5.1 category

```
constexpr Category Well::category {WELL}  [static], [constexpr], [private]
```

Menandakan bahwa land bertipe Well

The documentation for this class was generated from the following file:

- Cell/Well.h

4.39 World Class Reference

```
#include <World.h>
```

4.39.1 *

Public Member Functions

- World ()
- ∼World ()
- void Input ()
- void Update ()
- void Draw ()

4.39.2 *

Private Attributes

- Player pl
- Cell *** map
- int nRowCell
- int nCollumnCell
- LinkedList< FarmAnimal * > animalList

4.39.3 Constructor & Destructor Documentation

@ifstar

World()

4.39.3.1 World()

```
World::World ( )
```

Constructor World. Memanggil ctor dan menginisialisasi semua atribut world; Pertama, map diinisialisasi sesuai dengan spesifikasi, saat penginisialisasian map, ctor untuk object riil dari cell seperti coop, barn, dan well dipanggil Kedua, ctor Player dipanggil dengan argumen Point lokasi awal player dan reference ke map yang sudah didefinisikan pada tahap pertama Terakhir, animalList diinisialisasi dengan beberapa FarmAnimal secara random @ifstar

∼World()

4.39.3.2 ∼World()

```
World::∼World ( )
```

Destructor World. Dealokasi seluruh Cell dan FarmAnimal, termasuk seluruh pointer yang berhubungan.

4.39.4 Member Function Documentation

@ifstar

Draw()

4.39.4.1 Draw()

```
void World::Draw ( )
```

Megambarkar representasi state program (World) seperti lokasi setiap objek, money, water, dan Inventory Player, dsb ke layar. @ifstar

Input()

4.39.4.2 Input()

```
void World::Input ( )
```

Membaca input user dari stdin lalu melakukan aksi sesuai degan spesifikasi, misal, input == MOVELEFT, maka akan dipanggil pl.move(LEFT). Bila input == INTERACT, maka akan dipanggil pl.interact(animalList), dsb. @ifstar

Update()

4.39.4.3 Update()

```
void World::Update ( )
```

Pada World::Update(), setiap fungsi yang dipanggil secara berkala seperti FarmAnimal::tick() akan dipanggil.

4.39.5 Member Data Documentation

@ifstar

animalList

4.39.5.1 animalList

```
LinkedList<FarmAnimal*> World::animalList [private]
```

LinkedList dari seluruh pointer ke FarmAnimal yang berada pada World 000 @ifstar

map

4.39.5.2 map

```
Cell*** World::map [private]
```

Matriks dari pointer ke seluruh Cell pada World @ifstar

nCollumnCell

4.39.5.3 nCollumnCell

```
int World::nCollumnCell [private]
```

Nilai efektif kolom untuk Matriks Cell @ifstar

nRowCell

4.39.5.4 nRowCell

```
int World::nRowCell [private]
```

Nilai efektif baris untuk Matriks Cell @ifstar

pl

4.39.5.5 pl

```
Player World::pl [private]
```

Player yang berada pada World

The documentation for this class was generated from the following file:

- World.h

5 File Documentation

5.1 Cell/Barn.h File Reference

```
#include "Land.h"
```

5.1.1 *

Classes

- class Barn

5.2 Cell/Cell.h File Reference

5.2.1 *

Classes

- class Cell

5.3 Cell/Coop.h File Reference

```
#include "Land.h"
```

5.3.1 *

Classes

- class Coop

5.4 Cell/Facility.h File Reference

```
#include "Cell.h"
```

5.4.1 *

Classes

- class Facility

5.5 Cell/GrassLand.h File Reference

```
#include "Land.h"
```

5.5.1 *

Classes

- class GrassLand

5.6 Cell/Land.h File Reference

```
#include "Cell.h"
```

5.6.1 *

Classes

- class Land

5.7 Cell/Mixer.h File Reference

```
#include "Facility.h"
```

5.7.1 *

Classes

- class Mixer

5.8 Cell/Truck.h File Reference

```
#include "Facility.h"
```

5.8.1 *

Classes

- class Truck

5.9 Cell/Well.h File Reference

```
#include "Facility.h"
```

5.9.1 *

Classes

- class Well

5.10 Direction.h File Reference

5.10.1 *

Enumerations

- enum Direction { Direction::LEFT, Direction::RIGHT, Direction::UP, Direction::DOWN }

5.10.2 Enumeration Type Documentation

@ifstar

Direction

5.10.2.1 Direction

```
enum Direction  [strong]
```

Enumerator

LEFT	
RIGHT	
UP	
DOWN	

5.11 FarmAnimal/Chicken.h File Reference

```
#include "../Point.h"
#include "../Cell/Cell.h"
#include "EggProducer.h"
#include "MeatProducer.h"
#include <string>
```

5.11.1 *

Classes

- class Chicken

5.12 FarmAnimal/Cow.h File Reference

```
#include "../Point.h"
#include "../Cell/Cell.h"
#include "MilkProducer.h"
#include "MeatProducer.h"
#include <string>
```

5.12.1 *

Classes

- class Cow

5.13 FarmAnimal/Duck.h File Reference

```
#include "../Point.h"
#include "../Cell/Cell.h"
#include "MeatProducer.h"
#include <string>
```

5.13.1 *

Classes

- class Duck

5.14 FarmAnimal/EggProducer.h File Reference

```
#include "../Point.h"
#include "../Cell/Cell.h"
#include "FarmAnimal.h"
```

5.14.1 *

Classes

- class EggProducer

5.15 FarmAnimal/FarmAnimal.h File Reference

```
#include "../Point.h"
#include "../Cell/Cell.h"
#include "../LivingThing.h"
#include "../Product/FarmProduct.h"
#include <string>
```

5.15.1 *

Classes

- class FarmAnimal

5.16 FarmAnimal/Horse.h File Reference

```
#include "../Point.h"
#include "../Cell/Cell.h"
#include "MilkProducer.h"
#include <string>
```

5.16.1 *

Classes

- class Horse

5.17 FarmAnimal/MeatProducer.h File Reference

```
#include "../Point.h"
#include "../Cell/Cell.h"
#include "FarmAnimal.h"
```

5.17.1 *

Classes

- class MeatProducer

5.18 FarmAnimal/MilkProducer.h File Reference

```
#include "../Point.h"
#include "../Cell/Cell.h"
#include "FarmAnimal.h"
```

5.18.1 *

Classes

- class MilkProducer

5.19 FarmAnimal/Ostrich.h File Reference

```
#include "../Point.h"
#include "../Cell/Cell.h"
#include "EggProducer.h"
#include <string>
```

5.19.1 *

Classes

- class Ostrich

5.20 FarmAnimal/Sheep.h File Reference

```
#include "../Point.h"
#include "../Cell/Cell.h"
#include "MeatProducer.h"
#include <string>
```

5.20.1 *

Classes

- class Sheep

5.21 LinkedList.h File Reference

```
#include <initializer_list>
```

5.21.1 *

Classes

- class LinkedListNode< T >
- class LinkedList< T >
- class LinkedListNode< T >

5.22 LivingThing.h File Reference

```
#include "Point.h"
#include "Cell.h"
#include "Direction.h"
```

5.22.1 *

Classes

- class LivingThing

5.23 Player.h File Reference

```
#include "LivingThing.h"
#include "LinkedList.h"
#include "FarmAnimal.h"
#include "Cell/Cell.h"
#include "Product/BeefChickenOmelette.h"
#include "Product/BeefMuttonSate.h"
#include "Product/SuperSecretSpecialProduct.h"
#include "Point.h"
```

5.23.1 *

Classes

- class Player

5.24 Point.h File Reference

5.24.1 *

Classes

- struct Point

5.25 Product/BeefChickenOmelette.h File Reference

```
#include "../LinkedList.h"
#include "SideProduct.h"
```

5.25.1 *

Classes

- class BeefChickenOmelette

5.26 Product/BeefMuttonSate.h File Reference

```
#include "../LinkedList.h"
#include "SideProduct.h"
```

5.26.1 *

Classes

- class BeefMuttonSate

5.27 Product/ChickenEgg.h File Reference

```
#include "FarmProduct.h"
```

5.27.1 *

Classes

- class ChickenEgg

5.28 Product/ChickenMeat.h File Reference

```
#include "FarmProduct.h"
```

5.28.1 *

Classes

- class ChickenMeat

5.29 Product/CowMeat.h File Reference

```
#include "FarmProduct.h"
```

5.29.1 *

Classes

- class CowMeat

5.30 Product/CowMilk.h File Reference

```
#include "FarmProduct.h"
```

5.30.1 *

Classes

- class CowMilk

5.31 Product/DuckMeat.h File Reference

```
#include "FarmProduct.h"
```

5.31.1 *

Classes

- class DuckMeat

5.31.2 *

Macros

- #define DUCk_MEAT_H

5.31.3 Macro Definition Documentation

@ifstar

DUCk_MEAT_H

5.31.3.1 DUCk_MEAT_H

```
#define DUCk_MEAT_H
```

5.32 Product/FarmProduct.h File Reference

```
#include "Product.h"
```

5.32.1 *

Classes

- class FarmProduct

5.33 Product/HorseMilk.h File Reference

```
#include "FarmProduct.h"
```

5.33.1 *

Classes

- class HorseMilk

5.34 Product/OstrichEgg.h File Reference

```
#include "FarmProduct.h"
```

5.34.1 *

Classes

- class OstrichEgg

5.35 Product/Product.h File Reference

5.35.1 *

Classes

- class Product

5.36 Product/SheepMeat.h File Reference

```
#include "FarmProduct.h"
```

5.36.1 *

Classes

- class SheepMeat

5.37 Product/SideProduct.h File Reference

```
#include "Product.h"
```

5.37.1 *

Classes

- class SideProduct

5.38 Product/SuperSecretSpecialProduct.h File Reference

```
#include "../LinkedList.h"
#include "SideProduct.h"
```

5.38.1 *

Classes

- class SuperSecretSpecialProduct

5.39 World.h File Reference

```
#include "Player.h"
#include "LinkedList.h"
#include "Cell/Cell.h"
#include "FarmAnimal/FarmAnimal.h"
```

5.39.1 *

Classes

- class World

Index